

WHAT IS CLAIMED IS:

1 1. A method for preventing unauthorized access to hardware management
2 information comprising:
3 receiving a request for hardware component information in a service processor
4 disposed in a hardware component as an open session request from a requesting client
5 application, which request passed to the service processor external to an operating
6 system controlling the hardware component;
7 transmitting from the service processor a challenge string to the requesting client
8 application;

9 receiving in the service processor a challenge response from the requesting
10 client application;
11 comparing the challenge response to an expected response to the challenge
12 string; and

13 transmitting hardware component information to the requesting client
14 application.

1
1 2. The method according to claim 1, wherein the challenge string includes a
2 session identification number unique to each session.

1
1 3. The method according to claim 1, wherein the challenge response includes a
2 session identification number unique to each session and assigned by the service
3 processor.

1
1 4. The method according to claim 1, wherein the challenge response includes a
2 sequence number that increments with every new message.

1
1 5. The method according to claim 1, wherein the challenge response includes a
2 hash number, wherein the hash number is a function of one or more of the following:
3 the challenge string, the session identification number, the sequence number and a
4 password.

1
2 6. The method according to claim 1, further comprising examining each packet
3 received from the client application for one or more of the following: the session
identification number, the sequence number and a hash number.

1
2 7. The method according to claim 6, wherein the hash number is a function of
3 one or more of the following: the session identification number, the sequence number
and the packet itself.

1
2 8. A method for preventing unauthorized access to hardware management
information comprising:

3 transmitting a request for hardware component information to a service
4 processor disposed in a hardware component as an open session request from a
5 requesting client application;

6 passing the request to the service processor external to an operating system
7 controlling the hardware component;

8 receiving from the service processor a challenge string at the requesting client
9 application;

10 transmitting to the service processor a challenge response from the requesting
11 client application; and

12 receiving from the service processor an authentication response to the requesting
13 client application based on a comparison of the challenge response from the requesting
14 client application and an expected challenge response calculated in the service
15 processor.

1
2 9. The method according to claim 8, wherein the challenge string includes a
3 session identification number assigned by the service processor, which session
4 identification number is unique to each session, and the challenge response includes the
session identification number.

1

1 10. The method according to claim 9, wherein the challenge response includes a
2 sequence number that increments with every new message from the requesting client
3 application.

1

1 11. The method according to claim 8, wherein the challenge response includes a
2 hash number calculated by the requesting client application, and the hash number is a
3 function of one or more of the following: the challenge string, the session identification
4 number, the sequence number and a password.

1

1 12. The method according to claim 8, further comprising transmitting with
2 each packet sent by the client application one or more of the following: the session
3 identification number, the sequence number and a hash number, and the hash number is
4 a function of one or more of the following: the session identification number, the
5 sequence number and the packet itself.

1

1 13. An apparatus for authenticating a client application requesting access to a
2 particular component among a plurality of components, comprising:

3 a remote access port; and

4 a service processor disposed in the particular component, coupled to the remote
5 access port, and in response to a remote request for information about the particular
6 component received as an open session request through the remote access port external
7 to a host operating system, the service processor is programmed to:

8 transmit a challenge string to a requesting client application;

9 compare a challenge response received from the requesting client
10 application with an expected response to the challenge; and

11 transmit an authentication response to the requesting client application
12 based on the comparison.

1

1 14. The apparatus according to claim 13, wherein service processor assigns a
2 session identification number unique to each session and transmits the session
3 identification number to the requesting client application in the challenge string.

1

1 15. The apparatus according to claim 14, wherein the service processor reviews
2 the challenge response to determine if it contains the session identification number
3 transmitted in the challenge string.

1

1 16. The apparatus according to claim 13, wherein the service processor
2 compares a sequence number included in the challenge response against previously
3 received sequence numbers and ignores the challenge response if it does not include a
4 sequence number in correct sequence.

1

1 17. The apparatus according to claim 13, wherein the service processor
2 compares a hash number received in the challenge response with an expected hash
3 calculated by the service processor and transmits a success or failure message
4 depending upon a result of the comparison.

1

1 18. The apparatus according to claim 17, wherein the hash includes one or more
2 of the following: the challenge string, the session identification number, the sequence
3 number and a password.

1

1 19. The apparatus according to claim 13, wherein the service processor
2 examines each packet sent by the client application for one or more of the following:
3 the session identification number, the sequence number and a hash number, wherein the
4 hash number is a function of one or more of the following: the session identification
5 number, the sequence number and the packet itself.

1

1 20. A system for accessing hardware component information from a computer,
2 comprising:

3 a service processor disposed in the computer;

4 a server remotely coupled to the service processor in the computer;

5 a client application to execute on the server, wherein the service processor

6 authenticates requests from the client application requesting access to the service

7 processor's host hardware module, which request bypasses an operating system of the

8 computer, and the service processor in response to a request for access to the host
9 hardware module is programmed to:
10 transmit a challenge string to a requesting client application;
11 compare a challenge response received from the requesting client
12 application with an expected response to the challenge; and
13 transmit an authentication response to the requesting client application
14 based on the comparison.

1
1 21. The system according to claim 20, wherein each of the service processors
2 assigns a session identification number unique to each session and transmits the session
3 identification number to the requesting client application in the challenge string.

1
1 22. The system according to claim 20, wherein each of the service processors
2 reviews the challenge response to determine if it contains the session identification
3 number transmitted in the challenge string.

1
1 23. The system according to claim 20, wherein each of the service processors
2 compares a sequence number included in the challenge response against previously
3 received sequence numbers and ignores the challenge response if it does not include a
4 sequence number in correct sequence.

1
1 24. The system according to claim 20, wherein each of the service processors
2 compares a hash number received in the challenge response with an expected hash
3 calculated by the service processor and transmits a success or failure message
4 depending upon a result of the comparison.

1
1 25. The system according to claim 24, wherein the hash includes one or more of
2 the following: the challenge string, the session identification number, the sequence
3 number and a password.

1

1 26. The system according to claim 20, wherein each of the service processors
2 examines each packet sent by the client application for one or more of the following:
3 the session identification number, the sequence number and a hash number, wherein the
4 hash number is a function of one or more of the following: the session identification
5 number, the sequence number and the packet.

1
1 27. A method for verifying integrity of a data packet comprising:
2 receiving the data packet in a service processor disposed in a hardware
3 component from a client application, which data packet passes external to an operating
4 system and a system processor otherwise controlling operation of the hardware
5 component;

6 receiving with the data packet a keyed hash of the data packet; and
7 comparing the keyed hash with the data packet to an expected keyed hash.

1
1 28. The method according to claim 27, wherein the keyed hash is a function of
2 one or more of the following: a session identification number, a sequence number, a
3 password and the data packet.

1
1 29. A method for verifying integrity of a data packet comprising:
2 transmitting a data packet to a service processor disposed in a hardware
3 component from a client application, which data packet passes external to an operating
4 system and system processor otherwise controlling the hardware component;
5 calculating a keyed hash of the data packet; and
6 transmitting to the service processor the keyed hash along with the data packet.

1
1 30. The method according to claim 29, wherein the keyed hash is a function of
2 one or more of the following: a session identification number, a sequence number, a
3 password and the packet.

1 31. An apparatus for preventing unauthorized access to hardware management
2 information comprising a computer readable media having programming instructions
3 encoded thereon, instructing a processor to:

4 receive a request for hardware component information in a service processor
5 disposed in a hardware component as an open session request, which request passes
6 external to an operating system controlling the hardware component;

7 transmit from the service processor a challenge string to the requesting client
8 application;

9 receive in the service processor a challenge response from the requesting client
10 application;

11 compare the challenge response to an expected response to the challenge; and

12 transmit from the service processor an authentication response to the requesting
13 client application based on the comparison.

1
1 32. An apparatus for preventing unauthorized access to hardware management
2 information comprising a computer readable media having programming instruction
3 encoded thereon instructing a processor to:

4 transmit a request for hardware component information to a service processor
5 disposed in a hardware component as an open session request from a requesting client
6 application, which request passes external to an operating system controlling the
7 hardware component;

8 receive from the service processor a challenge string at the requesting client
9 application;

10 transmit to the service processor a challenge response from the requesting client
11 application; and

12 receive from the service processor an authentication response to the requesting
13 client application based on a comparison of the challenge response from the requesting
14 client application and an expected challenge response calculated in the service
15 processor.

1

- 1 33. An apparatus for verifying integrity of a data packet comprising a computer
- 2 readable media having programming instructions encoded thereon instructing a
- 3 processor to:
- 4 receive the data packet and a keyed hash in a service processor disposed in a
- 5 hardware component from a client application, which data packet and keyed hash pass
- 6 external to an operating system and a system processor otherwise controlling operation
- 7 of the hardware component;
- 8 calculate an expected a keyed hash of the data packet; and
- 9 compare the received keyed hash with the expected keyed hash.